JAAF: A Framework to Implement Self-Adaptive Agents Able to Deal with Web Services

Baldoino Fonseca
baldoino.fonseca@gmail.com
Roadmap

- Motivation
- OWL-S
- Model
  - Class Diagram
- Case Study:
  - GeoRisc
  - Adaptive Personal Web Page
- Related Work
- Conclusion
Motivation

- Service-oriented computing (SOC) has taken hold in business in, for instance, the use of shipping services in e-commerce transactions; the aggregation of hotel, car rental, and airline services.

- Therefore, it is necessary to provide techniques to discover, invoke, compose and monitor web services.

- Semantic Web Service (SWS) has been pointed as a way to address these issues.

- Although SWS can solve some of the mentioned issues, the complexity of current systems has directed the software engineering community to look for systems able to adjust or adapt their behavior in response to requirement changes.
Motivation

• Considering that adaptive agents present properties like: reasoning, learning, autonomy and pro-activity, multi-agent system is a paradigm that fits on these concerns.

• The Java self-Adaptive Agent Framework (JAAF) was proposed.
  – It extends the JADE framework;
  – It offers support to the implementation of different self-adaptation processes composed of activities that can perform:
    • Collect of data;
    • Analysis;
    • Decisions.
Motivation

- The Java self-Adaptive Agent Framework (JAAF)
  - It provides reasoning mechanisms based on:
    - Rules;
    - Cases;
    - Genetic Algorithm.
  - It offers selection mechanisms based on:
    - Utility Function;
    - Reputation.
  - It provides flexibility to create different plans for self-adaptation.
JAAF: Collect

- It is responsible for providing mechanisms to collect, aggregate and filter (format) data collected from the application.

- The collect has two sub-activities:
  - Sensor: It defines the place where the data should be collected (database, log, etc).
  - Format: It defines the format of the collected data.
The analyze activity is responsible for providing mechanisms that analyze the data collected in the previous activity in order to detect problems and suggest new solutions.

This activity gives support to three techniques:

- Rule-based reasoning (forward chaining, backward chaining and fuzzy logic) [12];
- Case-based reasoning [8];
• Decision
  – Decision is the activity responsible for deciding which action (or behavior) will be the next one to be executed by the agent, while trying to achieve the goal.
  – This activity gives support to two techniques:
    • Reputation;
    • Utility Function.

• Effector
  – It receives the selected action from the Decision activity, and informs the agent the action to be executed.
  – When the action is executed, the control loop can be executed again whether any self-adaptation is necessary.
The hot-spots specifically defined in JAAF are:

- **Agent (AdaptationAgent class):**
  - By extending such class and implementing the `executedPlan` method, it is possible to define different algorithms to execute the plans of an agent.

- **Plan of self-adaptation (PlanAdaptation class):**
  - It is possible to define new control loops (or plans) and the sequence to execute the activities of the control loops. JAAF already provides a default control loop implemented in the `ControlLoop` class.
JAAF: Hot-spots and Frozen-spots

- Activities (Behaviour class):
  - It is possible to define new activities to be called by the control loops by extending the Behaviour class. JAAF already offers four activities (Collect, Analyze, Decision and Effector).

- Sensor (Sensor class):
  - One can define when and where the data should be collected.

- Format (Format class):
  - It is possible to define the format of the data to be collected by the Sensor.

- Selection techniques:
  - Reputation and Utility Function.

- Intelligent Algorithm module:
  - JAAF offers three kinds of algorithms: rule-based reasoning (forward chaining, backward chaining and fuzzy logic), case-based reasoning and genetic algorithm.
GeoRisc
Problema

BR-101 – Rio-Santos

Quitandinha, Petrópolis, Dezembro de 2001

Morin, Petrópolis, Dezembro de 2002
Camadas
## Agente de Recomendação

### Regras

<table>
<thead>
<tr>
<th>Tipos de análises</th>
<th>Modelo</th>
<th>Características</th>
<th>Dados necessários</th>
<th>Escalas recomendadas</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Regional</td>
</tr>
<tr>
<td>Inventário</td>
<td>Distribuição dos escorregamentos</td>
<td>Análise da distribuição e classificação</td>
<td>3</td>
<td>sim</td>
</tr>
<tr>
<td></td>
<td>Atividade dos escorregamentos</td>
<td>Análise das mudanças do padrão temporal</td>
<td>4,5,14,15,16,17</td>
<td>não</td>
</tr>
<tr>
<td></td>
<td>Densidade dos escorregamentos</td>
<td>Cálculo da densidade nas unidades de terreno ou no mapa de</td>
<td>1,2,3</td>
<td>sim</td>
</tr>
<tr>
<td>Heurísticos</td>
<td>Geomorfológica</td>
<td>Usa observações de campo do perito no zoneamento</td>
<td>2,3,4</td>
<td>sim</td>
</tr>
<tr>
<td></td>
<td>Combinação qualitativa</td>
<td>Perito fornece ponderação de</td>
<td>2,3,5,6,7,8,9,10,12,14,16,18</td>
<td>sim</td>
</tr>
<tr>
<td>Estatísticas</td>
<td>Bivariada</td>
<td>Calcula a importância da combinação de</td>
<td>2,3,5,6,7,8,9,10,12,14,16,18</td>
<td>não</td>
</tr>
<tr>
<td></td>
<td>Multivariada</td>
<td>Calcula a fórmula de previsão a partir de uma matriz de</td>
<td>2,3,5,6,7,8,9,10,12,14,16,18</td>
<td>não</td>
</tr>
<tr>
<td>Determinísticas</td>
<td>Fator de segurança</td>
<td>Aplicada a modelos de hidrologia e estabilidade de</td>
<td>6,11,12,13,16,20,21,22,23</td>
<td>não</td>
</tr>
</tbody>
</table>
Regras Iniciais

| Geomorfologia | 1. Mapeamento de unidades de terreno  |
|               | 2. Unidades e subunidades geomorfológicas |
| Topografia    | 5. Modelo digital do terreno          |
|               | 6. Mapa de declividades              |
|               | 7. Mapa de direção dos taludes       |
|               | 8. Comprimento do talude             |
| Geotecnia     | 10. Litologia                        |
|               | 11. Pedologia                        |
|               | 12. Mapa geológico estrutural        |
| Uso do Solo   | 14. Infra-estrutura recente          |
|               | 15. Infra-estrutura antiga          |
|               | 16. Mapa de uso do solo recente      |
| Hidrologia    | 18. Drenagem                         |
|               | 19. Bacia hidrológica                |
|               | 20. Regime de chuvas                 |
|               | 21. Temperatura                      |
|               | 22. Evapotranspiração                |

<table>
<thead>
<tr>
<th>Caracterização</th>
<th>Escalas</th>
<th>Áreas mínimas de estudo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nacional</td>
<td>1:1.000.000</td>
<td>250.000 km²</td>
</tr>
<tr>
<td>Regional</td>
<td>1:100.000 a 1:500.00</td>
<td>2.500 a 62.500 km²</td>
</tr>
<tr>
<td>Média</td>
<td>1:25.000 a 1:50.000</td>
<td>56,25 a 625 km²</td>
</tr>
<tr>
<td>Grande</td>
<td>1:5.000 a 1:15.000</td>
<td>6,25 a 56,25 km²</td>
</tr>
<tr>
<td>Local</td>
<td>1:5.000 e maiores</td>
<td></td>
</tr>
</tbody>
</table>
Case Study: Motivation

- Landslides are natural phenomena, which are difficult to predict since they depend on many (unpredicted) factors and on relationships among those factors.

- One of the main challenges faced by specialists is to decide the most appropriate configuration of susceptibility model to generate susceptibility maps (SM).

- In this context, we used the JAAF framework to create a multi-agent system in order to generate a SM that shows the places with landslide risks of Rio de Janeiro, a city in Brazil.

- Each application agent is able to adapt the configuration of its susceptibility model in order to meet the SM that represents the reality.
Vision

- Interface Layer
- Generate Susceptibility Maps Layer
  - Manager Agent
  - Generate susceptibility maps
    - Inventory Agent
    - Heuristic Agent
    - Statistic Agent
    - Deterministic Agent
  - Web Services
Manager Agent
Adaptive Personal Web Pages
Related Works: A framework for Dynamic Adaptation of Power-Aware Server Clusters

PETRUCCI, V. ; LOQUES, O. ; Mossé, D. A framework for dynamic adaptation of power-aware server clusters.
Related Works: DySOA

Related Works: DySOA

Conclusion and Future Works

- New control-loops of self-adaptation
- Safe Adaptation
  - Norms
- Software Test and Metrics


The End!!

Questions?